Approximation of geometric set packing and hitting set problems

DOI:
10.26481/dis.20030924sk

Summary

This thesis develops approximation algorithms for several combinatorial optimization problems. These problems arise as special cases of the well known in combinatorial optimization set packing and hitting set problems, where the subsets in the input have a certain geometric structure. For illustration, consider the following examples:
Problem 1: given is a set of horizontal line segments placed arbitrarily in the plane. Find a maximum subset of segments the projections of which to the horizontal and vertical axes do not overlap mutually. See Figure 7.1 for illustration.

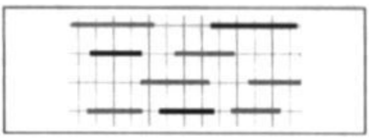

Figure 7.1: An instance of problem 1 and its optimal solution (shaded).

Problem 2: given is a set of horizontal line segments placed arbitrarily in the plane. Find a minimum number of horizontal and vertical lines, such that each segment is intersected by at least one line. Figure 7.2 illustrates this situation.

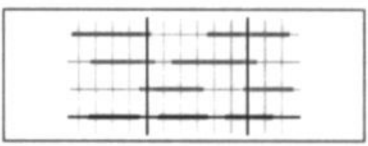

Figure 7.2: An instance of problem 2 and its optimal solution.

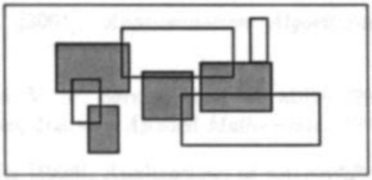

Figure 7.3: An instance of problem 3 and its optimum solution (shaded).

Problem 3: given is a set of axis-parallel closed rectangles of unit height and arbitrary length. Find the maximum number of non-overlapping ones. See Figure 7.3.

In this thesis we study the problems described above and more general versions of them including numerical parameters such as weights, demands and capacities. These problems find applications in molecular biology, printed circuit board manufacturing, cashing, load balancing, map labeling, database decision support and some other areas. Since all the studied problems are NP-hard, it is not realistic to find a polynomial (i.e, fast) algorithm that always returns an optimal solution. One of the ways is then to search for *approximation algorithms, i.e. polynomial algorithms*, which always find a solution of the value - for a minimization problem - at most δ times the value of an optimal solution ($\delta > 1$); for a maximization problem, the value of the solution obtain by the algorithm should be at least δ times the optimum value ($\delta < 1$). Parameter δ, called an approximation guarantee, determines how close are the algorithmic solutions to the optimum, and serves as a criterion of quality of an algorithm: the closer is δ to 1, the better. An algorithm with an approximation guarantee δ is called a δ-approximation algorithm.

Approximation algorithms developed in this thesis are based on different techniques. Since our problems allow for a natural integer programming formulation, such techniques as the primal-dual scheme and rounding of the linear programming relaxation (LP-rounding) can be used. Using the primal-dual scheme we develop approximation algorithms for two different generalizations of problem 1. These algorithms simultaneously construct a pair of feasible solution: one to a generalization of problem 1 and one to the related so-called dual problem, which is a corresponding generalization of problem 2. Each of those algorithms is shown to be a 1/2-approximation algorithm for problem 1 and a 2-approximation algorithm for problem 2 at the same time.

Further we study algorithms for problem 2 based on the LP-rounding technique. These algorithms prove in our case to provide better approximation factors than factor 2, provided by primal-dual algorithms. However, in contrast to primal-dual algorithms, LP-rounding algorithms have to solve a linear programming problem, which can be a time consuming although polynomial operation. We present several LP-rounding algorithms for different generalizations of problem 2. Moreover, we investigate whether or not and to which extant an approximation guarantee of our algorithms can be improved by another LP-rounding algorithm.

A polynomial-time approximation scheme (PTAS) for a maximization (minimization) problem is a family of δ-approximation algorithms, for each $\delta < 1$ ($\delta > 1$). In this thesis we show that no PTAS can exist for a certain class of problems, including problem 2, unless $\mathcal{P} = \mathcal{NP}$.

For problem 3, studied in the last chapter of this thesis, a polynomial time approximation scheme has been known before. It uses a so-called "shifting technique" in combination with a dynamic programming procedure. By designing a new dynamic programing procedure we improve the running time of this PTAS and its memory requirement.

tion.

Further we study algorithms for problem 2 based on the LP-rounding technique. These algorithms provide better approximation factors than factor 2, provided by primal-dual algorithms. However, in contrast to primal-dual algorithms, LP-rounding algorithms have to solve a linear programming problem, which can be a time consuming although polynomial operation. We present several LP-rounding algorithms for different generalizations of problem 2. Moreover, we investigate whether or not and to which extent an approximation guarantee of our algorithms can be improved by another LP-rounding algorithm.

A polynomial-time approximation scheme (PTAS) for a maximization (minimization) problem is a family of δ-approximation algorithms for each $\delta < 1$ ($\delta > 1$). In this thesis we show that no PTAS can exist for a certain class of problems, including problem 2, unless P = NP.

For problem 3, studied in the last chapter of this thesis, a polynomial time approximation scheme has been known before. It uses a so-called 'shifting' technique in combination with a dynamic programming procedure. By designing a new dynamic programming procedure we improve the running time of the PTAS and its memory requirement.

Samenvatting

Dit proefschrift beschrijft approximatie-algoritmen voor verscheidene combinatorische optimaliseringsproblemen. De bestudeerde problemen zijn speciale gevallen van het bekende 'set-packing' probleem en het 'hitting-set' probleem; de speciale gevallen kenmerken zich door een geometrische structuur die in de invoer aanwezig is. Beschouw, ter illustratie de volgende voorbeelden.

Probleem 1: gegeven is een verzameling van horizontale lijnstukken, willekeurig in het vlak geplaatst. Vind nu een zo groot mogelijke deelverzameling van lijnstukken zodanig dat de doorsnede van de projecties van elk der lijnstukken zowel op de horizontale as als op de verticale as leeg is (zie Figuur 8.1).

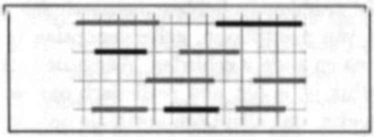

Figure 8.1: Een instantie van probleem 1 en de bijbehorende optimale oplossing (in grijs).

Probleem 2: gegeven is een verzameling van horizontale lijnstukken, willekeurig in het vlak geplaatst. Vind nu een minimaal aantal horizontale en verticale lijnen zodanig dat elk lijnstuk door tenminste één lijn doorsneden wordt (zie Figuur 8.2).

Probleem 3: gegeven is een verzameling van rechthoeken, parallel met de assen, die elk hoogte 1, en een willekeurige lengte hebben. Vind nu een maximaal aantal elkaar niet-overlappende rechthoeken (zie Figuur 8.3).

In dit proefschrift bestuderen we dergelijke problemen en hun generalisaties naar gewogen versies (waarbij een geselecteerd lijnstuk of rechthoek een gegeven gewicht oplevert) en/of naar versies met een gegeven vraag